

Observe que ainda temos três comandos para refinar:

- "houver dados"
- "cargo estiver correto" e
- "não produziu mais de 130 ou não tem gratificações"

Para o primeiro, poderíamos ter

```
refinamento de "houver dados"
| FUNCIONÁRIO · INSCRIÇÃO ≠ 0 {flag}
|
| fim refinamento.
```

Para o segundo, poderemos ter

```
refinamento de "cargo estiver correto"
| "Verificar se o salário da tabela para o cargo é igual ao salário do funcionário";
|
| fim refinamento.
```

Precisamos definir a tabela de cargos

```
tipo v = vetor [1..N] t;
tipo t = registro
| inteiro: CARGO,
| real: SALÁRIO.
```

```

| fim registro;
v TABELADECARGOSESALÁRIOS
t CARGOSESALÁRIOS.
```

Vamos fazer a carga da tabela via cartões:

```
leia CAR, SAL;
enquanto CAR ≠ 0 faça
| i ← i + 1;
| TABELADECARGOSESALÁRIOS[i] · CARGO ← CAR;
| TABELADECARGOSESALÁRIOS[i] · SALÁRIO ← SAL;
| leia ICAR, SAL;
| fim enquanto;
N ← i {nº de entradas da tabela efetivamente utilizadas}
```

refinamento de "verificar se o salário da tabela para o cargo é igual ao salário do funcionário"

```

CARGOCORRETO ← falso;
para i de 1 até N faça
| se FUNCIONÁRIO · CARGO =
| TABELADECARGOSESALÁRIOS[i] · CARGO
| então
| se FUNCIONÁRIO · SALÁRIO =
| TABELADECARGOSESALÁRIOS[i] · SALÁRIO
| então
| CARGOCORRETO ← verdadeiro;
| abandone;
| senão
```

```

| CARGOCORRETO ← falso;
| abandone;
| fim se;
| fim para;
| fim refinamento.
```

```

refinamento de "não produziu mais de 130 ou não tem gratificações"
| FUNCIONÁRIO · QUANTIDADEPRODUZIDA ≥ 130
| e FUNCIONÁRIO · TOTALGRATIFICAÇÕES ≥ 0;
|
| fim refinamento.
```

Finalmente o algoritmo completo

```

algoritmo
| tipo t = registro
| | inteiro: INSCRIÇÃO
| | inteiro: CARGO,
| | caractere: NOME,
| | real: SALÁRIO,
| | inteiro: QUANTIDADEPRODUZIDA,
| | real: TOTALGRATIFICAÇÕES;
| fim registro;
| t FUNCIONÁRIO;
| inteiro N, i, real PAGAMENTOBRUTOMENSAL;
| lógico CARGOCORRETO;
| tipo v = vetor [1..1000] t;
| tipo t = registro
| | inteiro: CARGO,
| | real: SALÁRIO;
| fim registro;
| v TABELADECARGOSESALÁRIOS;
| t CARGOSESALÁRIOS;
| inteiro {módulo de carga da tabela};
| inteiro CAR;
| real SAL;
| leia (CAR, SAL);
| i ← 0;
| enquanto CAR ≠ 0 faça
| | i ← i + 1;
| | TABELADECARGOSESALÁRIOS[i] · CARGO ← CAR;
| | TABELADECARGOSESALÁRIOS[i] · SALÁRIO ← SAL;
| | leia (CAR, SAL);
| fim enquanto;
| N ← i; {número de entradas utilizadas}
| fim;
```